Vignette: Immunosuppression, rather than inflammation, is a salient feature of sepsis in an Indian cohort

Samanwoy Mukhopadhyay[*] *and Saroj Kant Mohapatra*[†]

21 August, 2019

Contents

1 General summary

Sepsis is one of the leading diseases associated with high mortality and morbidity, due to the systemic nature of this illness, transcriptomic technology is particularly suited to investigation of molecular underpinning of survival from sepsis episodes. We adopted an analysis approach that combined published transcriptome data and data generated in our laboratory from Indian sepsis patients leading to the discovery of key immune pathways to be altered in non-survivors compared to survivors. This is the first clinical transcriptomic study on sepsis from India, showing that non-survival is associated with down-regulated adaptive immune pathways and significant M2-specific immune-suppression, possibly regulated by NF-κB signalling. Three Biological processes related to sepsis were observed to be significantly altered in non-survivors. A patient-specific analysis reveals up-regulation of coagulation and inflammation but a strong down-regulation of immunosuppression modules in Indian sepsis patients.

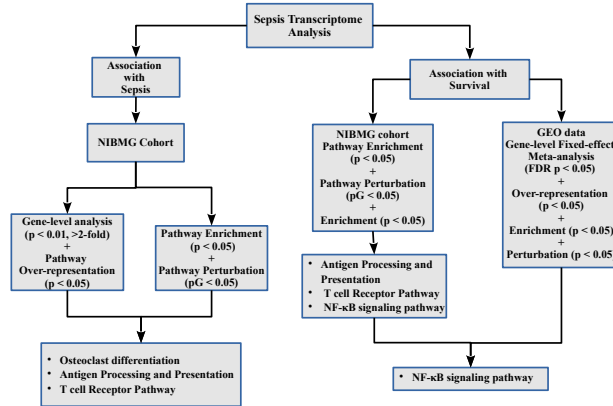[*]sm3@nibmg.ac.in
[†]skm1@nibmg.ac.in

Figure 1: Analysis flow with results: The left arm describes the steps for case-vs-control analysis whereas the right arm describes the steps for identification of pathways associated with survival

2   Getting the data

Availability of data and materials Data and R code are available at the following link (https://figshare.com/) (search for the project **ssnibmgsurv**). The data are accessed through the two data packages listed below. The code can be downloaded as a single zip file (**ssnibmgsurvdoc.zip**). Upon uncomperssing the zip, install the two data packages as described below and run the subsequent code to generate the appropriate output.

2.1   Installation of the data package ssnibmgsurv

Dowload the file **ssnibmgsurv_1.0.tar.gz** from https://figshare.com/ (search for **ssnibmgsurv**). Change the directory to where you saved the file. Start R. At the R prompt, issue the following command:

```
install.packages(pkgs="ssnibmgsurv_1.0.tar.gz", repos=NULL)
# Now the data package ssnibmgsurv is installed on your computer.
# Check with the following command:
library("ssnibmgsurv")
```

2.2   Installation of the data package ssgeosurv

Dowload the file **ssgeosurv_1.0.tar.gz** from https://figshare.com/ (search for **ssgeosurv**). Change the directory to where you saved the file. Start R. At the R prompt, issue the following command:

```
install.packages(pkgs="ssgeosurv_1.0.tar.gz", repos=NULL)
# Now the data package ssgeosurv is installed on your computer.
# Check with the following command:
library("ssgeosurv")
```

Now both the data packages are installed in your computer; let's focus on starting the analysis.

2.3   Running the analysis: Setting working directory

It is assumed that you have access to a folder **ssnibmgsurvdoc**. Start R and set the working directory to **ssnibmgsurvdoc**. Run the code chunks as follow.

3 Case control analysis: Identifying genes and pathways DE in sepsis

```r
# Clearing the workspace and close any graphics window if open
rm(list=ls())
graphics.off()
# Loading the preliminary libraries including data packages
source("Rcode/prelim.R")
```

```r
source("Rcode/getData.R")
# Using age- and gender-matched controls
matched.12 <- c("C11","C8","C1","C7","C17","C10","C21","C18","C20","C4","C9",
                "C12","42D1","1D1","8D1","50D1","60D1", "90D1","62D1","70D1",
                "19D1","32D1","14D1","61D1")
esetm <- eset[, matched.12]
rttm <- rowttests(esetm, "Group")
lfcdm<-rttm$dm
pdm <- p.adjust(rttm$p.value, method="BH")
names(pdm) <- rownames(rttm)
names(lfcdm) <- rownames(rttm)
egs.all <- featureNames(esetm)
# Removing some variables that are not to be used for further analysis
rm(snames, ptids)
```

```r
# Draw a volcano plot to show that 24% of the genome
# are perturbed in sepsis (FDR p < 0.05)
# sepsis and matched control samples are used here
source("Rcode/makeplot_1_volcano_for_genomic_storm.R")
```
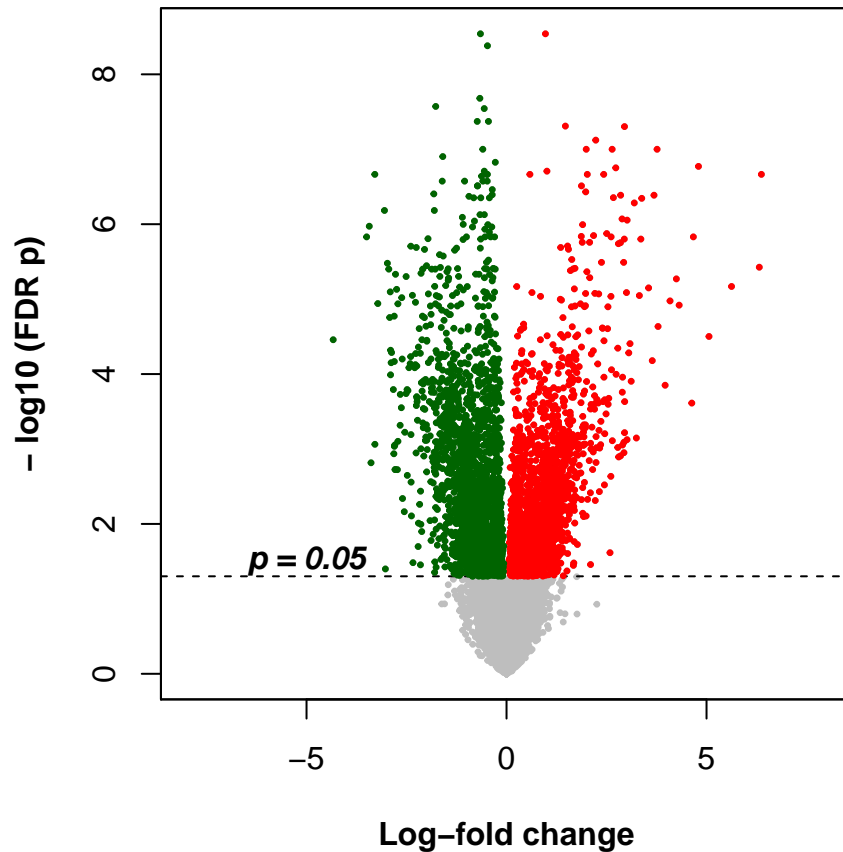
Figure 2: Volcano plot showing 24 percent of the genome perturbed in sepsis compared to healthy control (FDR p < 0.05). This establishes large scale change in gene expression in sepsis, and possible multiple pathways being perturbed.

3.1 Genome-level changes in gene expression

Differential gene expression analysis revealed 1109 genes to be altered in sepsis patients compared to age- and gender-matched healthy controls. Volcano plot (Figure 2) showed 24% of the genome perturbed in sepsis compared to healthy control (FDR p < 0.05). This establishes large scale change in gene expression in sepsis, and possible multiple pathways being perturbed.

```
# Detect the highly significant DE genes of sepsis - FDR p < 0.01; 2-fold
# Draw box-plot to show temporal changes in control vs cases
upg <- egs.all[which(pdm<0.01 & lfcdm>1)]
downg <- egs.all[which(pdm<0.01 & lfcdm<(-1))]
deg.d1<-union(upg,downg)

# Line plot showing slow return or non-survivors to baseline gene expression
source("Rcode/makeplot_2_temporal_plot_for_DEgns.R")
```

Figure 3: Temporal change of DE genes (FDR p < 0.05, 2 fold-change or more), there is a non-random trend toward the baseline with time (p-values from paired t-tests are provided in the legend). This is consistent with earlier findings from patients with trauma.

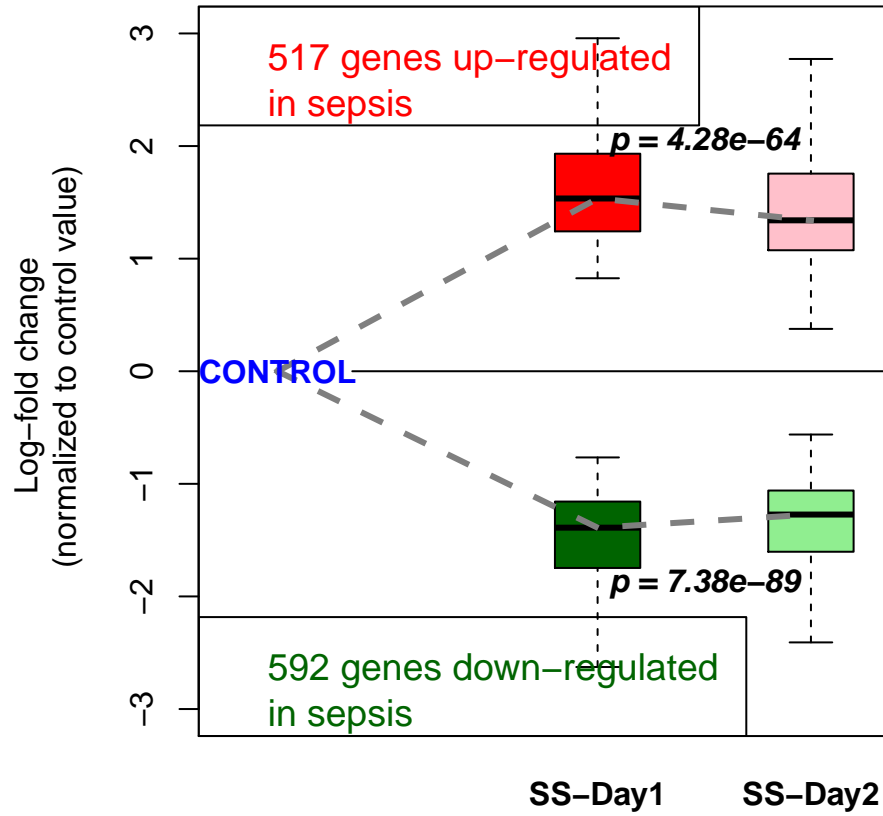**Temporal progression of 1109 differentially regulated genes**
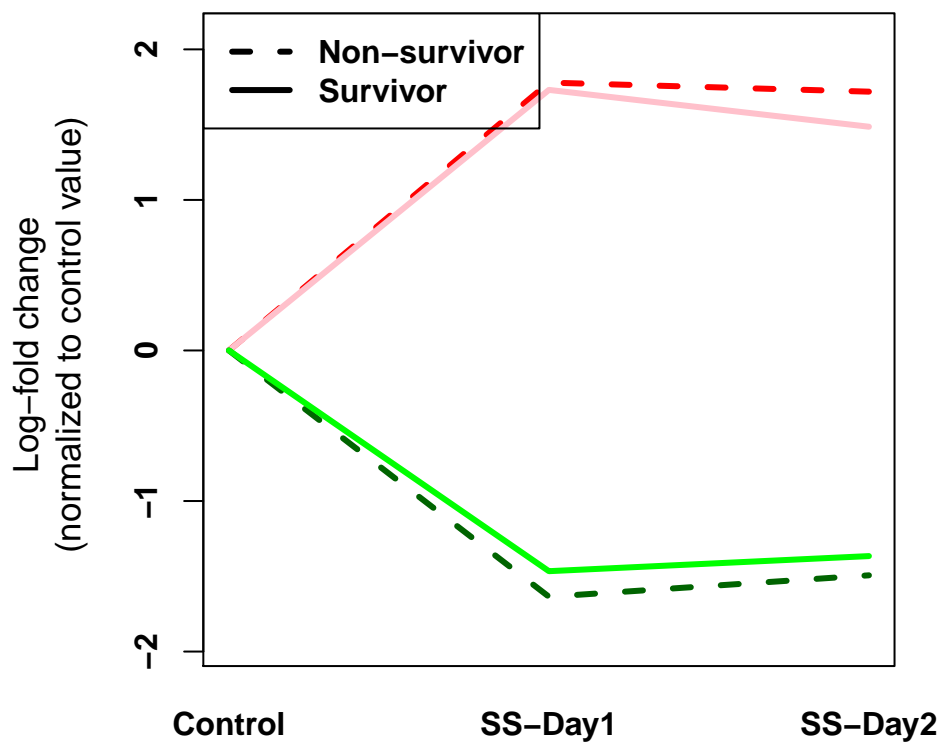
Figure 4: Temporal change of DE genes (FDR p < 0.05, 2 fold-change or more), there is a non-random trend toward the baseline with time (p-values from paired t-tests are provided in the legend). The delayed return to baseline is associated with non-recovery from sepsis.

```
# draw trajectory of DE genes survivor versus non-survivor
source("Rcode/drawDEtrajectorySurvival.R")
```

3.2 Pathway analysis results

```r
# Pathway Analysis: ORA, GSEA, SPIA
# ORA - Over representation analysis
pORAup <- getORApvals(upg, egs.all)[,"p"]
pORAup <- p.adjust(pORAup, method="BH")
pORAdown <- getORApvals(downg, egs.all)[,"p"]
pORAdown <- p.adjust(pORAdown, method="BH")

NIBMG.ORA.disease <- cbind(pORAup, pORAdown)
colnames(NIBMG.ORA.disease) <- paste(colnames(NIBMG.ORA.disease),
                                     "NIBMG.disease", sep="_")

# GSEA - Gene Set Enrichment Analysis
# SPIA - Signaling Pathway Impact Analysis
# Warning running this code will take long time (approx 10 minutes each)
source("Rcode/run_GSEA_sepsis_vs_control.R") # GSEA
```

```
##
##  Loading GSEA permutation t.test result from file ...
##   done!
```

```r
source("Rcode/run_SPIA_sepsis_vs_control.R") # SPIA
```

```
##  Loading SPIA
##  result from file ...  done!
```

```r
# Combine result from 3 pathway analyses and print the Down and Up pathways#
pathsDown = intersect(intersect(names(which(pGSEAdown < 0.01)),
                            names(which(pORAdown < 0.01))), names(which(pG< 0.01)))
pathways.list[ paste0("path:", pathsDown)]
```

```
##                                            path:hsa03013
##                "RNA transport - Homo sapiens (human)"
##                                            path:hsa04612
## "Antigen processing and presentation - Homo sapiens (human)"
##                                            path:hsa04660
##   "T cell receptor signaling pathway - Homo sapiens (human)"
##                                            path:hsa05332
##        "Graft-versus-host disease - Homo sapiens (human)"
```

```r
# Removing some variables that are not to be used for further analysis
rm(pathsDown)

pathsUp = intersect(intersect(names(which(pGSEAup < 0.01)),
                          names(which(pORAup < 0.01))), names(which(pG< 0.01)))
pathways.list[ paste0("path:", pathsUp)]
```

```
##                                            path:hsa04380
##          "Osteoclast differentiation - Homo sapiens (human)"
##                                            path:hsa05133
##                     "Pertussis - Homo sapiens (human)"
##                                            path:hsa05150
##      "Staphylococcus aureus infection - Homo sapiens (human)"
##                                            path:hsa05202
## "Transcriptional misregulation in cancer - Homo sapiens (human)"
```

```
##                                                                path:hsa05322
##              "Systemic lupus erythematosus - Homo sapiens (human)"
```

```r
# Removing some variables that are not to be used for further analysis
rm(pathsUp)
```

Figure 5: Temporal boxplot of 3 key pathwayes altered in Sepsis by combined pathway analysis.

```r
par(mfrow=c(1,3))
# Box plot of two pathways down-regulated in sepsis
box.plot.KEGG(id="hsa04612", direction="down") # Antigen processing and presentation
box.plot.KEGG(id="hsa04660", direction="down") # T cell receptor signaling
# Box plot up-regulated pathway: Osteoclast Differentiation
box.plot.KEGG(id="hsa04380", direction="up")
```

4 Survival analysis

4.1 Survival analysis using published transcriptome data (from NCBI GEO)

```r
# Section B: Survival analysis
# Getting genes and pathways associated with survival
# uses NIBMG and published data sets
##################################################
rm(list=ls())
graphics.off()

##########################################################
# Preliminries
source("Rcode/prelim.R")
```
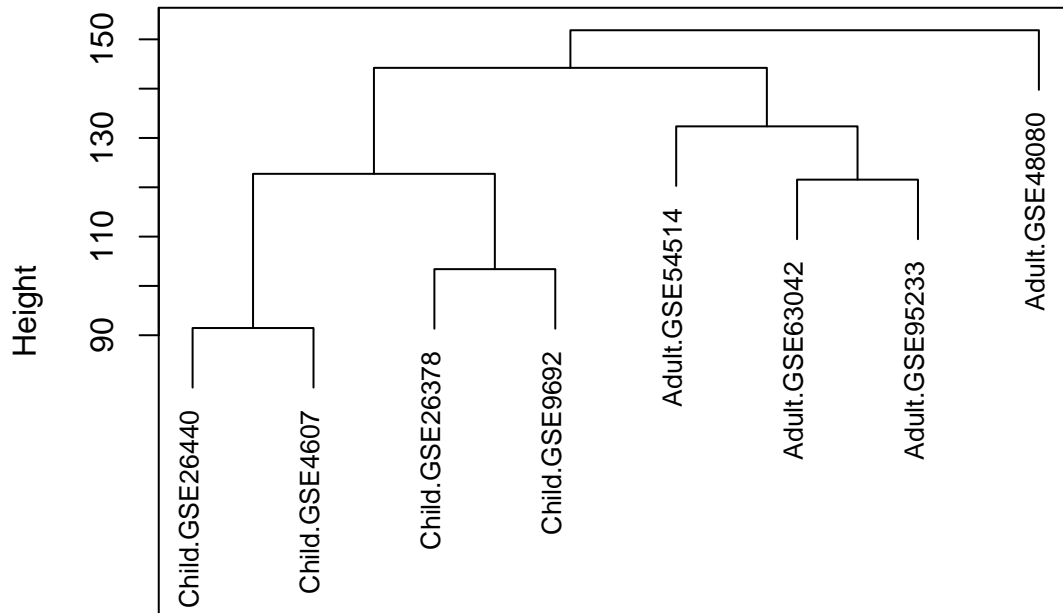
Figure 6: Survival analysis with eight published data sets: human adults and children with sepsis; hierarchical clustering with log-fold change in gene expression led to evidence of developmental age-specific differential perturbation; i.e., separate clusters for adult and child data sets. In view of this difference, further analysis was confined to to Adult data sets when combined with NIBMG data.

```
###############################
# Get the expression set
source("Rcode/getData.R")
library(ssgeosurv)
data(ss.list) # eight data sets = 4 adult + 4 child
data(ss.surv.list) # 8 datasets with survivors and non-survivors
studies = read.table(file="metadata/studies.txt", header=TRUE, sep="\t")
study.ids = as.character(studies$study.id)
study.type = as.character(studies$age)
names(study.type) = study.ids

##########################################
# Day 1 non-survivor vs survivor with FDR p cutoff 0.01
rtt1 <-rowttests(eset.s, factor(eset.s$Outcome))
sel1 <- rownames(rtt1)[which(rtt1$p.value < 0.01)]
sel1.nibmg.lfc <- rtt1[sel1,]

# Hierarchical clustering of log-fold change of 8 studies
source("Rcode/run_hclust.R")
box()
```

```r
# Pathway analysis in GEO data
# ORA: over-representation Analysis
source("Rcode/run_ORA_surv_nonsurv_analysis.R")
# Perform permutation-based GSEA analysis
source("Rcode/run_GSEA_surv_nonsurv_analysis.R")
```

```
## Loading gsea_child data from file ... done!
## File exists ...
## Done!...
## Loading gsea_adult data from file ... done!
## File exists ...
## Done!...
```

```r
##############################
# Two-way evidence plot adult
source("Rcode/run_SPIA_surv_nonsurv_analysis_adult.R")
```

```r
res <- spia.res[[1]]
resall.adult <-  data.frame(rep(names(ss.adult)[1],  nrow(res)),
                            res$Name, res$ID, res$NDE,
                            res$pNDE, res$tA, res$pPERT,
                            res$pG, res$pGFdr, res$Status)

col.nm <- as.character(sapply(strsplit(colnames(resall.adult), "res."), "[[", 2))
col.nm[1] <- "Study"

colnames(resall.adult)<- col.nm

for(i in 2:length(ss.adult)) {
  res <- spia.res[[i]]
  resedited <- data.frame(rep(names(ss.adult)[i],
                          nrow(res)), res$Name, res$ID, res$NDE, res$pNDE,
                          res$tA, res$pPERT, res$pG, res$pGFdr, res$Status)

  colnames(resedited)<- col.nm
  resall.adult <- rbind(resall.adult, resedited, deparse.level=0)
}

############################################################
# Calculate Fisher's product of p-values of pertabation for all pathways
keggs <- as.character(unique(resall.adult$ID))

# Create empty vector for capturing fisher product of pG
pPERT.Fp.adult <- vector(mode="numeric", length=length(keggs))
names(pPERT.Fp.adult) <- keggs

for(id in keggs) {
  pvec <- resall.adult[resall.adult$ID==id, "pPERT"]
  pPERT.Fp.adult[id] <- Fisher.test(pvec)["p.value"]

}
rm(pvec)

# Combine by Fisher product the two p values
# for perturbation (pb) and hypergeometric test (ph)
pb <- pPERT.Fp.adult
pb <- p.adjust(pb, "fdr")
ph <- as.numeric(pNDE.paths.ad[paste("hsa",names(pb),sep=""),"p"])
names(ph) <- names(pb)

# Use a floor value for p
ph[ph < 1e-07] <- 1e-07
pb[pb < 1e-07] <- 1e-07

pGmeta.adult <- combfunc(pb,ph, "fisher")

# Capture the fisher product of pPERT pG Meta p values into a dataframe
fisher.prod.spia.adult <- as.data.frame(cbind(paste("hsa",
                                          names(pPERT.Fp.adult), sep=""),
                                      as.numeric(ph), as.numeric(pPERT.Fp.adult),
```

```r
                                      as.numeric(pGmeta.adult)))
colnames(fisher.prod.spia.adult) <- c("paths", "pNDE.adult",
                                      "pPERT.adult", "pG.meta.adult")


#####################
# Following code is derived from SPIA::plotP
# The pG threshold is the p-value 0.05 corrected for the number of
# pathways being considered
tr= 0.05
#tr<-  0.05/length(pb)

# plot neg.log.p_PERT against neg.log.p_NDE
plot(-log(ph), -log(pb), col="gray80",
     xlim = c(0, max(c(-log(ph), -log(pb)) +1, na.rm = TRUE)),
     ylim = c(0, max(c(-log(ph), -log(pb) +1), na.rm = TRUE)),
     pch = 19, main = "Two-way evidence plot : Adult Sepsis", cex = 1.5,
     xlab = "Evidence of Over-representation, -log(p_ORA)",
     ylab = "Evidence of Perturbation, -log(p_PERT)")


# For selected  pathways for visualisation: NLR, NFkB, Osteoclast
###################################
sel.paths.ad <- c("04621", "04064", "04380")
col.vec <- c("red2", "purple2", "darkblue")
points(-log(ph)[sel.paths.ad ], -log(pb)[sel.paths.ad ], pch = 19, col = col.vec,
       cex = 1.5)
abline(v = -log(tr), lwd = 1, col = "red", lty = 2)
abline(h = -log(tr), lwd = 1, col = "red", lty = 2)
path.nms <- as.character(sapply(strsplit(
  pathways.list[paste("path:hsa", sel.paths.ad, sep="")],
  " -"), "[[", 1))
# Add a legend to the plot
legend("topright",  title="Pathway Names",
       legend= paste(sel.paths.ad , path.nms, sep="  "),
       text.font=2, text.col = col.vec, horiz=F,
       cex=0.8, pch=20, col= col.vec)
```
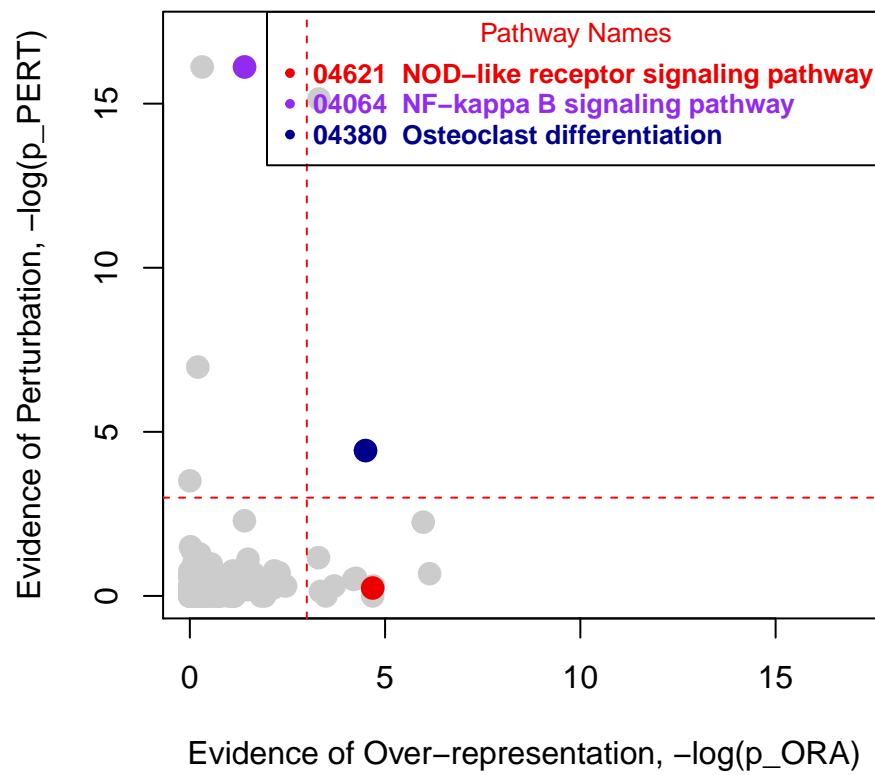
Figure 7: KEGG pathways associated with survival in GEO data; NF-kappaB signalling pathway, Osteoclast differentiation and NOD-like receptor signalling pathway.
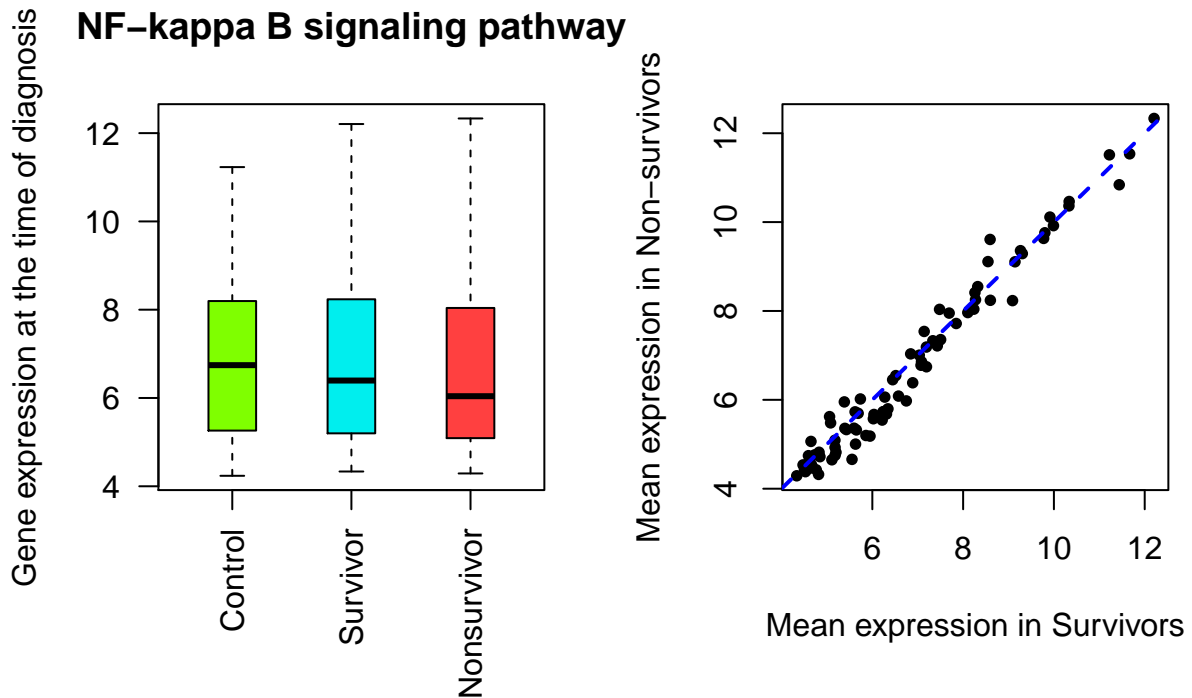
Figure 8: NF-kappa B signalling pathway boxplot and scatterplot

4.2 Down-regulation of NF-κ B signalling pathway genes in non-survivors

```r
rm(list=ls())
############################################################
# Preliminries
source("Rcode/prelim.R")

# Get the expression set
source("Rcode/getData.R")

# show gene expression trend for NF-kB signaling pathway
getPval(keggid="hsa04064", drawPlot=TRUE, getSigGenes=TRUE)
```

```
##  [1] "TNFRSF13C" "CSNK2A2"   "ICAM1"
##  [4] "LCK"       "BCL2A1"    "RELB"
##  [7] "BCL2L1"    "TRAF5"     "BCL10"
## [10] "CD40"
```

```r
# Draw the permutation histogram of GSEA for NF-kB signaling pathway
drawPermutHist(keggid="hsa04064", eset=eset.s, fac=factor(as.character(eset.s$Outcome)))
```
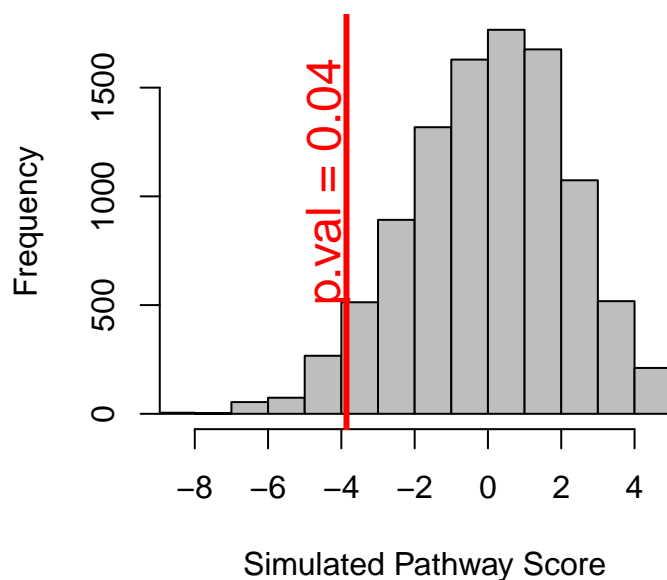
```
##   p.val    zobs
## 0.0441 -3.8605
```

Figure 9: NF-kappa B signalling pathway histogram. Permutation bases Gene set enrichment analysis creats a histogram of simulated pathway scores (in gray bars). The red line shows the observed pathway score in nonsurvivors when compared to survivor.

4.3 Relative gene expression of the targets of NF-κB

```
################################################################################
# drawing a histogram for NFkB targets
################################################################################
source("Rcode/plotDensityNFkbTargets.R")
```
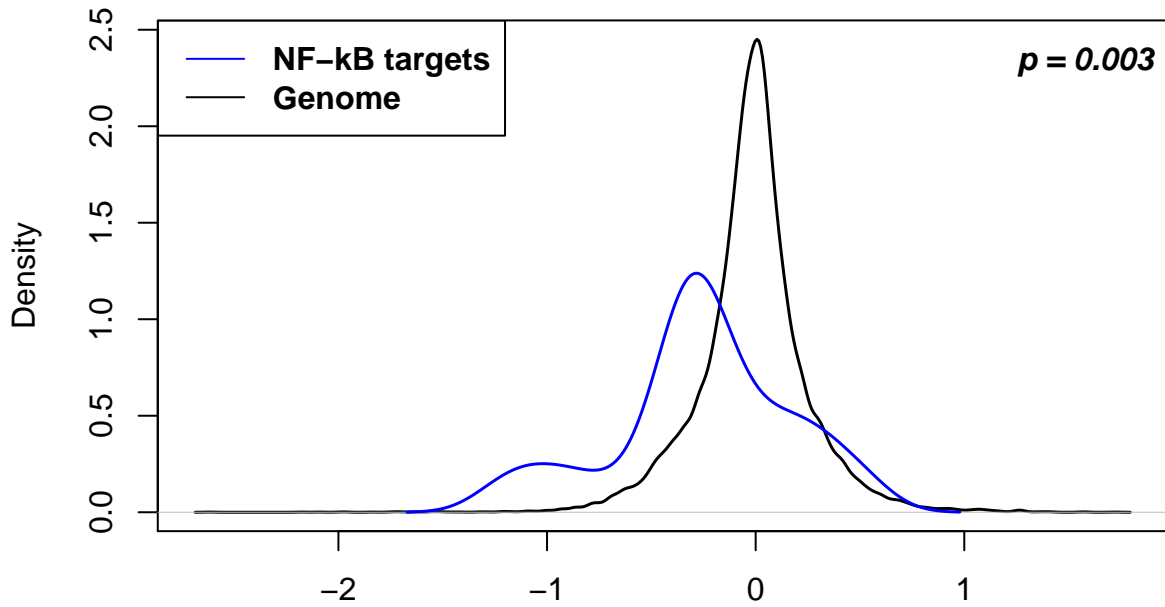
Figure 10: Relative gene expression of the targets of NFkB (i.e., Antigen processing and presentation genes and Immune receptor Genes). The gray peak in the background representsdistribution of all genes in the genome. There is significant down-regulation of the targets in the non-survivors (blue line).

4.4 M2 macrophage-specific down-regulation of gene expression in non-survivors

```r
# check the expression of M1 vs M2 markers in data from SCB cohort
# read the file containing gene IDs
###############
sel.gns.dat <- read.table("metadata/M1_M2_markers.txt", sep="\t", header=T)
m1.gns <- sel.gns.dat[,1]
m1.gns <- intersect(m1.gns, featureNames(eset))
m2.gns <- sel.gns.dat[,2]
m2.gns <- intersect(m2.gns, featureNames(eset))


# function for plotting macrophage-specific gene expression
plotMgexp = function(type="M1", normalizeByControl=FALSE) {
  if(type=="M1") {
    egs = m1.gns
  } else {
    egs = m2.gns
  }

  # M1 gene expression
  gexp.s = rowMeans(exprs(eset[egs, eset$Outcome=="Surv" & eset$Group=="D1"]))
  gexp.ns = rowMeans(exprs(eset[egs, eset$Outcome=="Nonsurv" & eset$Group=="D1"]))

  if(normalizeByControl==TRUE) {
    gexp.c = rowMeans(exprs(eset[egs, which.ctrl]))
    gexp.s  = gexp.s-gexp.c
    gexp.ns = gexp.ns-gexp.c
  }
```

```r
  plot(x = gexp.s, y=gexp.ns, las=1, cex=0.62, col= "blue", pch=16,
       ylab = "Mean expression in non-survivors", xlab = "Mean expression in survivors",
       main=paste0(type, "-specific gene expression"))
  abline(0,1, lty=2, xlim= c(-3, 10), ylim=c(-3, 10))
  gsyms <- as.character(unlist(mget(m1.gns, org.Hs.egSYMBOL)))
  #gsyms[intersect(which(gexp.1[,1] < -1), which(gexp.1[,2] < -1) )] <- ""
  text(x = gexp.s, y=gexp.ns, labels=gsyms, cex= 0.58, pos=2, offset = 0.75, font=2)

  pval <- t.test(gexp.s, gexp.ns, paired = T)$p.value
  legend.str <- paste("p = ", formatC(pval, digits=1), sep="")
  legend("topleft", legend.str, bty="n", text.font=4)
}
```
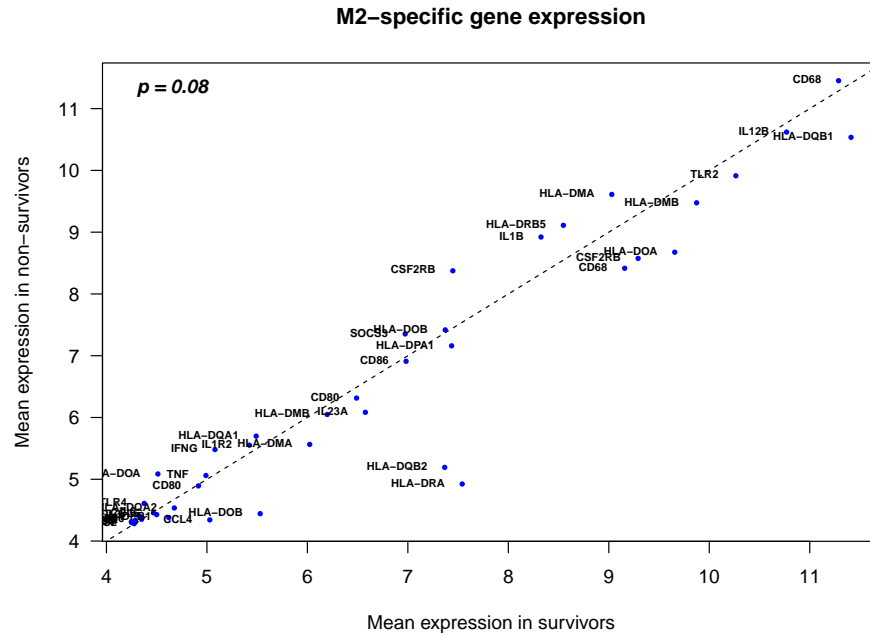
M2–specific gene expression

Figure 11: M1 macrophages (classically activated macrophages) are pro-inflammatory, important in host defence against the pathogens, phagocytosis, secretion of pro-inflammatory cytokines and microbicidal molecules. M2 macrophages (alternatively activated macrophages) participate in regulation of resolution of inflammation and repair of damaged tissues. M2-specific under-expression is observed in non-survivors (p = 0.02).

```
plotMgexp("M2")
```

4.5 Down-regulation of Antigen processing and presentation signalling pathway genes in non-survivors

```
###################################################
# Survivor versus non-survivor
###################################################
getPval("hsa04612", drawPlot=TRUE, getSigGenes=TRUE) # AgPP
```

```
##  [1] "HLA-DPA1" "HLA-DPB1" "HLA-DRB1"
##  [4] "HLA-DRB5" "HSPA5"    "KIR2DL3"
##  [7] "KIR3DL2"  "CIITA"    "PSME1"
## [10] "TAP1"     "TAP2"     "CD74"
```
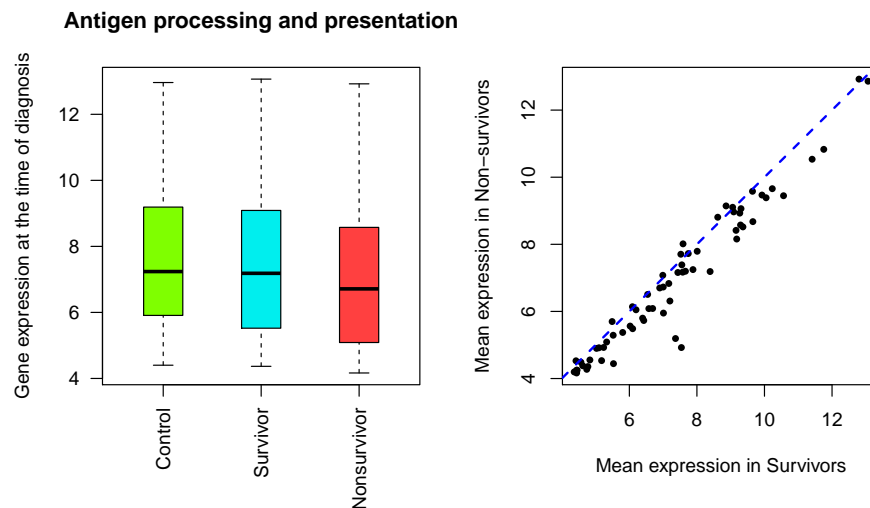
Antigen processing and presentation

Figure 12: Status of AgPP pathway in NIBMG data with Box/Scatterplot

```
drawPermutHist("hsa04612", eset=eset.s, fac=factor(as.character(eset.s$Outcome)))# AgPP
```
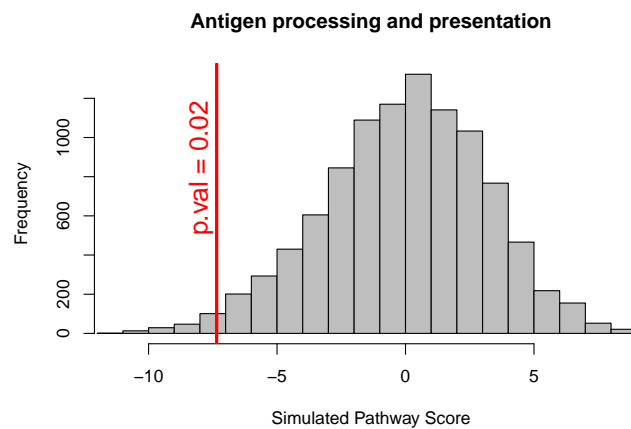
```
##   p.val   zobs
## 0.0154 -7.3492
```

Antigen processing and presentation

Figure 13: Status of AgPP pathway in NIBMG data, permutation based Gene set enrichment histogram.

4.6 Down-regulation of T cell receptor signalling pathway genes in non-survivors

```
####################################################
# Survivor versus non-survivor
####################################################
getPval("hsa04660", drawPlot=TRUE, getSigGenes=TRUE) # TCR
```

```
## [1] "RASGRP1" "ITK"     "LCK"
## [4] "NFKBIE"  "SOS2"    "BCL10"
## [7] "CD3D"    "CD3E"    "CD3G"
```
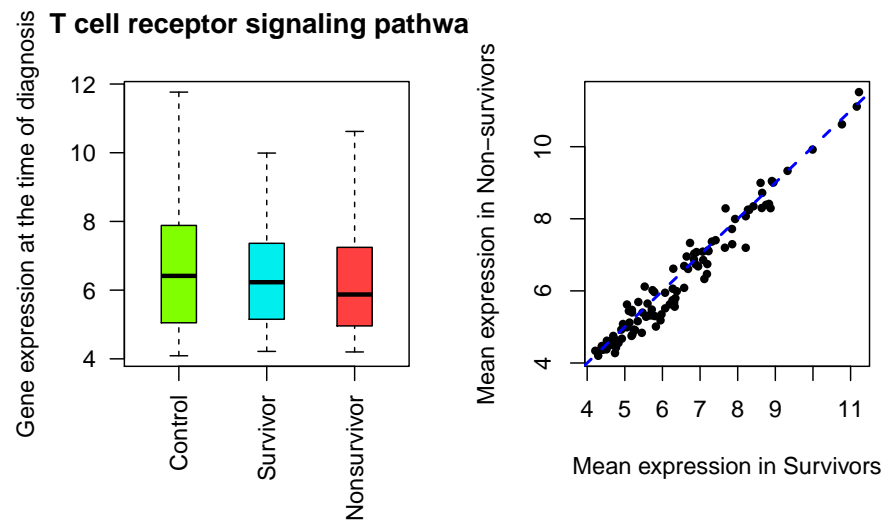
Figure 14: Box plot and scatter plot showing down-regulation of TCR pathway in non-survivors

```
drawPermutHist("hsa04660", eset=eset.s, fac=factor(as.character(eset.s$Outcome)))# TCR
```

```
##  p.val   zobs
##  0.114 -2.671
```


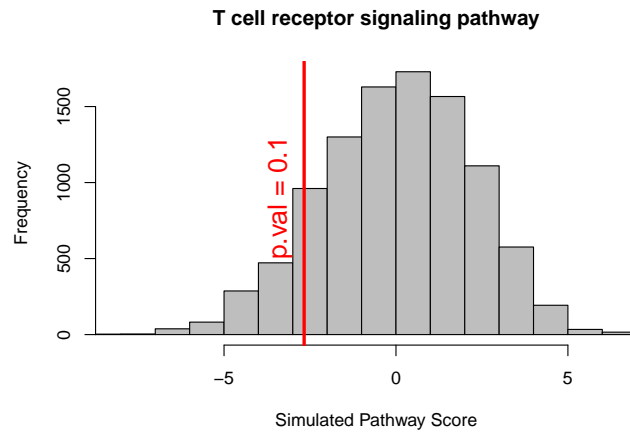**T cell receptor signaling pathway**

Figure 15: Down-regulation of TCR pathway in SCB cohort, with histogram from permutation based Gene set enrichment test.

```
net.dat <- read.table("metadata//network_input.csv", sep="\t", header=T)
net.list = with(net.dat, split(x=Egid, f=Process))

par(mfrow=c(3, 1), mar=c(2, 16, 2, 16))
for(i in 1:3) {
  drawPermutHist(geneids=net.list[[i]],
                 titlestr=names(net.list)[i],
                 fac=eset.s$Outcome,
                 eset=eset.s, cex= 0.80)
  box()
}
```
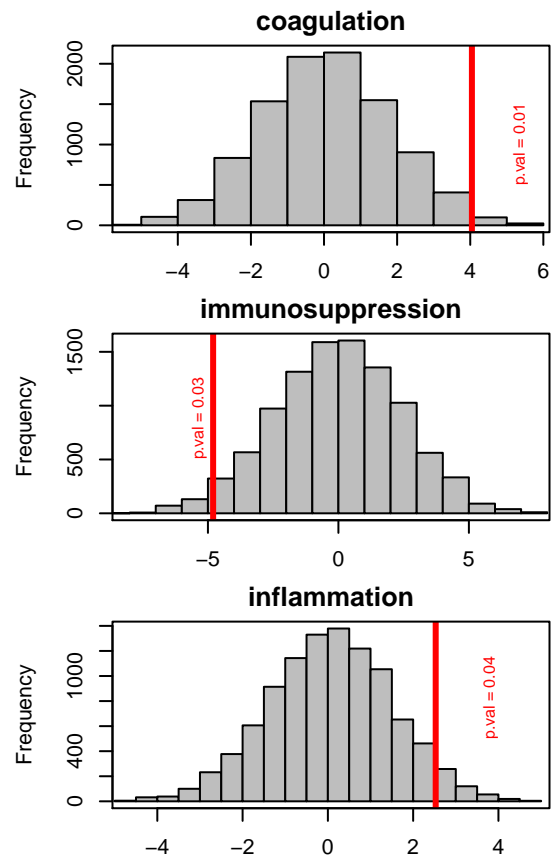
Figure 16: Three Biological processes found to be differentially enriched in Nonsurvivors. Refer to the main manuscript for more details.

5 Module scores in SCB cohort

```r
# get scores for the three modules from NIBMG data
fn = "Results/nibmgModuleScores.rda"
if(file.exists(fn)) {
  cat("Reading NIBMG module scores from file ...")
  load(fn)
  cat(" done!\n")
} else {
  sids = sampleNames(eset)[-c(which.ctrl)]
  ids.con = sampleNames(eset)[which.ctrl]
  modScores = sapply(names(net.list), function(id) {
    egs = as.character(net.list[[id]])
    sapply(sids, function(sid) {
      eset.curr = eset[,c(sid, ids.con)]
      eset.curr$Group = factor(as.character(eset.curr$Group))
      rtt = rowttests(eset.curr,"Group")
      rttstats = rtt[egs, "statistic"]
      z = sum(rttstats)/sqrt(length(egs))
      return(z)
    })
  })
  save(modScores, file=fn)
}
```

```
## Reading NIBMG module scores from file ... done!
```

```r
# format the data
sids = rownames(modScores)
sids.df = do.call(rbind,strsplit(sids, split="D"))
colnames(sids.df) = c("Pt","Day")
rm(sids)
dat = data.frame(sids.df, sids=rownames(modScores), modScores)

dat.split = with(dat, split(sids, Pt))
dat.split = sapply(dat.split, as.character)

par(mar=c(3,5,2,1))
dat = sapply(dat.split, function(x)
      as.matrix(dat[unlist(x),
      c("coagulation","immunosuppression","inflammation")]))
dat1 = dat
for(i in 1:length(dat)) {dat1[[i]] = rbind(dat[[i]], c(0,0,0))}
o = order(sapply(dat, function(x) min(x)))
dat1 = dat1[o]
plotdat = t(do.call(rbind, dat1))
mycols = c("blue", "darkgreen", "red")
b = barplot(plotdat, width=1, ylim=c(-15, 15), beside = T,
        col=mycols, , ylab="Module score",
        border=mycols, las=2, cex.names=0.9)
legend("top", c("Coagulation", "Immunosuppression", "Inflammation"), horiz=T,
      col=mycols,
      border=mycols,
      pch=15, pt.cex= 1.7, bty="n", inset=c(-0.05, 0))
```
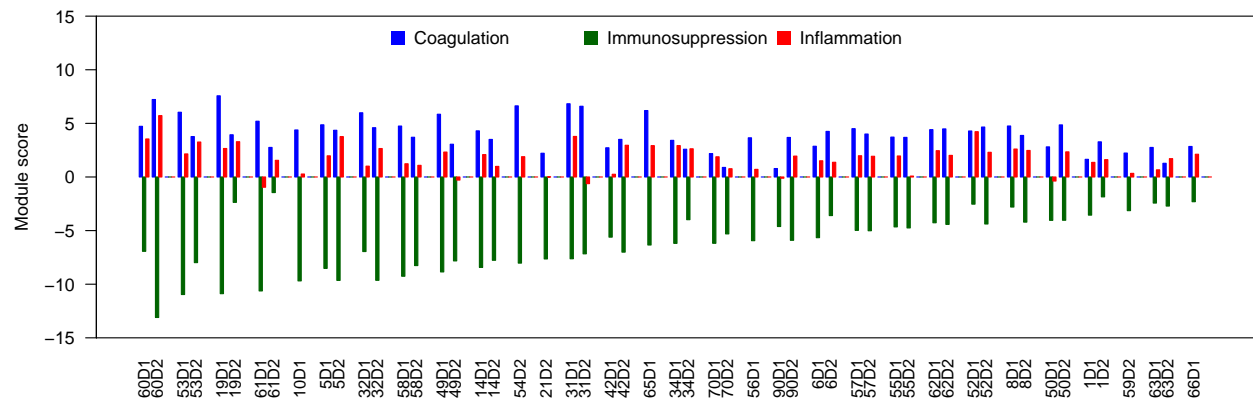
Figure 17: Barplot showing the score of 3 key modules in each sepsis patient

```
box()
```

```r
require(ggplot2)
dat <- t(abs(plotdat))
apply(t(abs(dat)), 1, median)
```

```
##       coagulation immunosuppression
##             2.807             3.983
##      inflammation
##             0.768
```

```r
df <- dat
sem <- function(x){
  sd(x)/sqrt(length(x))
}
my_mean <- apply(df, 2, mean)
my_sem <- apply(df, 2, sem)
# new data frame for storing the mean and sem
mean_sem_old <- data.frame(means=my_mean, sems=my_sem, group=colnames(df))
mean_sem <- rbind(mean_sem_old[2,], mean_sem_old[1,], mean_sem_old[3,])
rownames(mean_sem) < c("A", "B", "C")
```

```
## [1] FALSE FALSE FALSE
```

```r
# larger font
theme_set(theme_gray(base_size = 2))
# factorize the variable for legend
mean_sem$group <- factor(mean_sem$group, levels = mean_sem$group)
ggplot(mean_sem, aes(x=group, y=means, fill=group)) + theme(legend.position="right") +
  geom_bar(stat='identity', width=0.4) +
  geom_errorbar(aes(ymin=means-sems, ymax=means+sems), width=.12) +
    scale_fill_brewer(palette="Dark2")+
    xlab('')  + theme(legend.text=element_text(size=12, face="bold")) +
    theme(legend.title = element_blank())+
    ylab('Absolute module score (z)') +
    theme(plot.title = element_text(color="red", size=18, face="bold"),
          axis.title.x = element_text(color="blue", size=12, face="bold"),
          axis.title.y= element_text(color="blue", size=12, face="bold"))+
  theme(axis.text.x = element_text(size=0.0012)) +
  theme(axis.text.y = element_text(size=11, face="bold")) +
  geom_hline(yintercept=0, color="black", size= 0.46) +
  geom_vline(xintercept=0.61, color="black", size= 0.46)
```
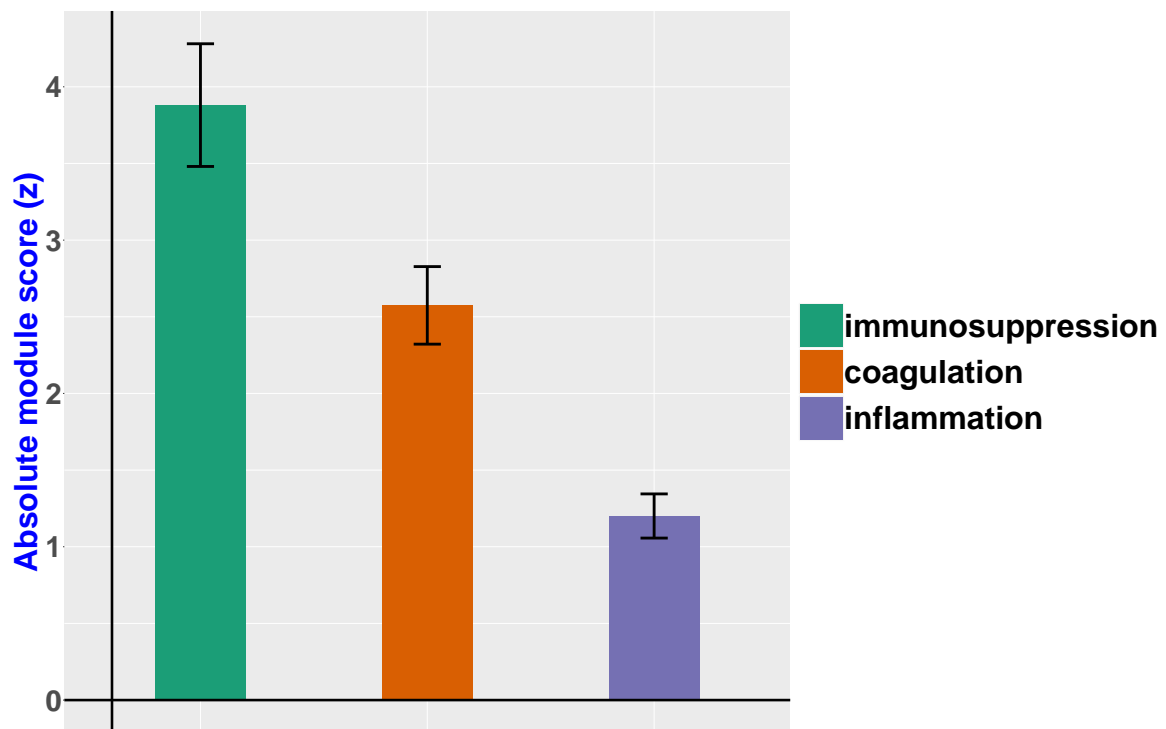
Figure 18: Barplot showing the magnitude of 3 key modules in each sepsis patients. Each bar represents mean of patient-level module score with SEM as the error bar. For all three modules, absolute z-scores have been used. It is clear that there is much greater immunnosuppression compared to inflammtion.

6 Session Information

```
## R version 3.4.4 (2018-03-15)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.1 LTS
##
## Matrix products: default
## BLAS: /usr/lib/x86_64-linux-gnu/openblas/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/libopenblasp-r0.2.20.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8
##  [2] LC_NUMERIC=C
##  [3] LC_TIME=en_IN.UTF-8
##  [4] LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_IN.UTF-8
##  [6] LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_IN.UTF-8
##  [8] LC_NAME=C
##  [9] LC_ADDRESS=C
## [10] LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_IN.UTF-8
## [12] LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats4    parallel  stats
## [4] graphics  grDevices utils
## [7] datasets  methods   base
##
## other attached packages:
##  [1] ssgeosurv_1.0
##  [2] ssnibmgsurv_1.0
##  [3] ggplot2_3.1.0
##  [4] KEGG.db_3.2.3
##  [5] pathview_1.18.2
##  [6] stringr_1.4.0
##  [7] gplots_3.0.1
##  [8] pca3d_0.10
##  [9] rgl_0.99.16
## [10] KEGGREST_1.18.1
## [11] SPIA_2.30.0
## [12] KEGGgraph_1.38.0
## [13] Category_2.44.0
## [14] Matrix_1.2-15
## [15] GSEABase_1.40.1
## [16] graph_1.56.0
## [17] annotate_1.56.2
## [18] XML_3.98-1.19
## [19] illuminaHumanv2.db_1.26.0
## [20] hgu133plus2.db_3.2.3
## [21] org.Hs.eg.db_3.5.0
## [22] AnnotationDbi_1.40.0
## [23] IRanges_2.12.0
## [24] S4Vectors_0.16.0
```

```
## [25] genefilter_1.60.0
## [26] limma_3.34.9
## [27] GEOquery_2.46.15
## [28] Biobase_2.38.0
## [29] BiocGenerics_0.24.0
##
## loaded via a namespace (and not attached):
##  [1] bitops_1.0-6
##  [2] bit64_0.9-7
##  [3] RColorBrewer_1.1-2
##  [4] webshot_0.5.1
##  [5] httr_1.4.0
##  [6] Rgraphviz_2.22.0
##  [7] tools_3.4.4
##  [8] R6_2.4.0
##  [9] KernSmooth_2.23-15
## [10] lazyeval_0.2.1
## [11] colorspace_1.4-0
## [12] DBI_1.0.0
## [13] manipulateWidget_0.10.0
## [14] withr_2.1.2
## [15] tidyselect_0.2.5
## [16] bit_1.1-14
## [17] compiler_3.4.4
## [18] xml2_1.2.0
## [19] labeling_0.3
## [20] caTools_1.17.1.1
## [21] scales_1.0.0
## [22] readr_1.3.1
## [23] RBGL_1.54.0
## [24] digest_0.6.18
## [25] rmarkdown_1.13
## [26] XVector_0.18.0
## [27] pkgconfig_2.0.2
## [28] htmltools_0.3.6
## [29] htmlwidgets_1.3
## [30] rlang_0.3.1
## [31] RSQLite_2.1.1
## [32] shiny_1.2.0
## [33] bindr_0.1.1
## [34] jsonlite_1.6
## [35] crosstalk_1.0.0
## [36] gtools_3.8.1
## [37] dplyr_0.7.8
## [38] RCurl_1.95-4.12
## [39] magrittr_1.5
## [40] Rcpp_1.0.0
## [41] munsell_0.5.0
## [42] stringi_1.4.3
## [43] yaml_2.2.0
## [44] zlibbioc_1.24.0
## [45] plyr_1.8.4
## [46] grid_3.4.4
## [47] blob_1.1.1
```

```
## [48] gdata_2.18.0
## [49] promises_1.0.1
## [50] crayon_1.3.4
## [51] miniUI_0.1.1.1
## [52] lattice_0.20-38
## [53] Biostrings_2.46.0
## [54] splines_3.4.4
## [55] hms_0.4.2
## [56] knitr_1.23
## [57] pillar_1.3.1
## [58] codetools_0.2-16
## [59] glue_1.3.1
## [60] evaluate_0.14
## [61] png_0.1-7
## [62] httpuv_1.4.5.1
## [63] gtable_0.2.0
## [64] purrr_0.2.5
## [65] tidyr_0.8.2
## [66] assertthat_0.2.0
## [67] xfun_0.8
## [68] mime_0.6
## [69] xtable_1.8-3
## [70] later_0.7.5
## [71] survival_2.43-3
## [72] tibble_2.0.1
## [73] memoise_1.1.0
## [74] ellipse_0.4.1
## [75] bindrcpp_0.2.2
```